Refine Search

Search Results -

Term	Documents
SELECT$	0
SELECT	320239
SELECTA	63
SELECTABBLE	1
SELECTABE	2
SELECTABEL	7
SELECTABILILTY	1
SELECTABILITY	962
SELECTABILITY/ACCESSIBILITY	1
SELECTABLE	74087
SELECTABLEATTENUATION	1
(L15 AND (SELECT$ WITH THREAD$ WITH AVAILAB$)).USPT.	0

There are more results than shown above. Click here to view the entire set.

Database:
US Pre-Grant Publication Full-Text Database
US Patents Full-Text Database
US OCR Full-Text Database
EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Search: L16

Refine Search

Recall Text Clear Interrupt

Search History

DATE: Monday, March 01, 2004 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
	DB=USPT; PLUR=YES; OP=ADJ		

h e b b cg b e e ch

L16	L15 and (select$ with thread$ with availab$)	0	L16
L15	L14 and (select$ with thread$ with process$ with packet$)	6	L15
L14	709/$.ccls.	14530	L14
L13	L12 and (select$ with thresd$)	0	L13
L12	(6463527 or 6366998).pn.	2	L12
L11	L10 and (subroutine$)	1	L11
L10	4075691.pn.	1	L10
L9	L8 and interface.ab.	21	L9
L8	L7 and controller$.ab.	204	L8
L7	davis$.in.	17643	L7
L6	L5 and (subroutine$)	0	L6
L5	4016548.pn.	1	L5
L4	L2 and port$.ab.	11	L4
L3	L2 and port$.ab.	11	L3
L2	L1 and (multiplexer$).ab.	32	L2
L1	law$.in.	36007	L1

END OF SEARCH HISTORY

h e b b cg b e e ch

First Hit Fwd Refs

Generate Collection

L15: Entry 1 of 6 File: USPT Sep 23, 2003

DOCUMENT-IDENTIFIER: US 6625654 B1
TITLE: Thread signaling in multi-threaded network processor

Current US Original Classification (1):
709/230

Current US Cross Reference Classification (1):
709/200

Current US Cross Reference Classification (2):
709/245

CLAIMS:

1. A method for network packet processing comprises: receiving network packets; and operating on the network packets with a plurality of program threads to affect processing of the packets; wherein the plurality of program threads are scheduler program threads to schedule task orders for processing and processing program threads that process packets in accordance with task assignments assigned by the scheduler program threads, the scheduler program threads can schedule any one of a plurality of processing program threads to handle processing of a task, the scheduler program thread writes a register with an address corresponding to a location of data for the plurality of processing program threads, a selected one of the plurality of processing program threads that can handle the task reads the register to obtain the location of the data, the selected one of the plurality of processing program threads reads the register to obtain the location of the data and to assign itself to processing the task requested by the scheduler program thread.

10. A method for network packet processing comprises: receiving network packets; and operating on the network packets with a plurality of program threads to affect processing of the packets; wherein the plurality of program threads are scheduler program threads to schedule task orders for processing and processing program threads that process packets in accordance with task assignments assigned by the scheduler program threads, the scheduler program threads can schedule any one of a plurality of processing program threads to handle processing of a task, the scheduler program thread writes a register with an address corresponding to a location of data for the plurality of processing program threads, a selected one of the plurality of processing program threads that can handle the task reads the register to obtain the location of the data, the selected one of the plurality of processing tasks reads the register to obtain the location of the data, while the register is cleared by reading the register by the program thread to assign itself to process the task.

h e b b g eeef e e g

US006625654B1

(12) **United States Patent**
Wolrich et al.

(10) **Patent No.: US 6,625,654 B1**
(45) **Date of Patent: Sep. 23, 2003**

(54) **THREAD SIGNALING IN MULTI-THREADED NETWORK PROCESSOR**

(75) Inventors: **Gilbert Wolrich**, Framingham, MA (US); **Debra Bernstein**, Sudbury, MA (US); **Donald Hooper**, Shrewsbury, MA (US); **Matthew J. Adiletta**, Worc, MA (US); **William Wheeler**, Southboro, MA (US)

(73) Assignee: **Intel Corporation**, Santa Clara, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/473,799**

(22) Filed: **Dec. 28, 1999**

(51) **Int. Cl.**[7] **G06F 15/16**
(52) **U.S. Cl.** **709/230**; 709/200; 709/102; 709/245
(58) **Field of Search** 709/200, 102, 709/245, 230; 713/155

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,627,829	A *	5/1997	Gleeson et al.	370/230
5,689,566	A *	11/1997	Nguyen	713/155
5,742,782	A *	4/1998	Ito et al.	712/210
5,983,274	A *	11/1999	Hyder et al.	709/230
6,085,215	A *	7/2000	Ramakrishnan et al.	709/102
6,212,542	B1 *	4/2001	Kahle et al.	709/102

OTHER PUBLICATIONS

Schmidt et al, "The Preformance of Alternative Threading Architectures for Parallel Communication Subsystems," Internet Document, Online! Nov. 13, 1998.

Vibhatavanijt et al., "Simultaneous Multithreading–Based Routers"Proceedings of the 2000 International Conference on Parallel Processing, Toronto, Ontario, Canada. Aug. 21–24, 2000, pp. 362–369.

Turner, Jonathan et al., "Design of a High Performance Active Router," Internet Document, Online Mar. 18, 1999.

Gomez, J.C. et al., "Efficient Multithreaded User–Space Transport for Network Computing: Design and Test of the Trap Protocol,"Journal of Parallel and Dsitributed Computing, Academic Press, Duluth, MN, US, vol. 40, No. 1, Jan. 10, 1997 pp. 103–117.

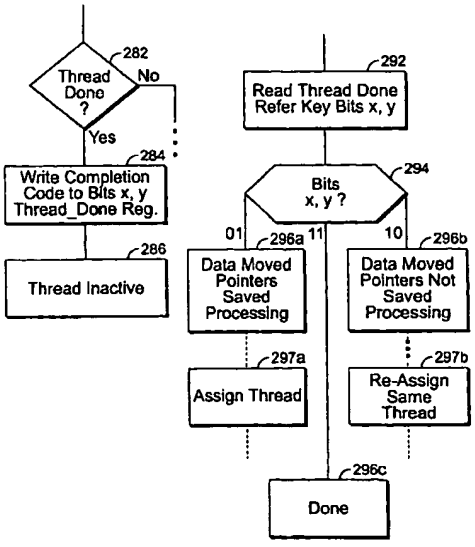
* cited by examiner

Primary Examiner—David Wiley
Assistant Examiner—Phuoc H. Nguyen
(74) *Attorney, Agent, or Firm*—Fish & Richardson P.C.

(57) **ABSTRACT**

A parallel hardware-based multithreaded processor is described. The processor includes a general purpose processor that coordinates system functions and a plurality of microengines that support multiple program threads. The processor also includes a memory control system that has a first memory controller that sorts memory references based on whether the memory references are directed than even bank or an odd bank of memory and a second memory controller that optimizes memory references based upon whether the memory references are read references or write references. A program thread communication scheme for packet processing is also described.

16 Claims, 17 Drawing Sheets



First Hit Fwd Refs

Generate Collection

L15: Entry 2 of 6 File: USPT Nov 19, 2002

DOCUMENT-IDENTIFIER: US 6483804 B1
TITLE: Method and apparatus for dynamic packet batching with a high performance network interface

Detailed Description Text (286):
In state 718, a thread or other process on the selected processor begins processing the packet that was stored in the processor's queue. Methods of processing a packet through its protocol stack are well known to those skilled in the art and need not be described in detail. The illustrated procedure then ends with end state 720.

Current US Cross Reference Classification (2):
709/225

Current US Cross Reference Classification (3):
709/228

US006483804B1

(12) **United States Patent**
Muller et al.

(10) **Patent No.: US 6,483,804 B1**
(45) **Date of Patent: Nov. 19, 2002**

(54) **METHOD AND APPARATUS FOR DYNAMIC PACKET BATCHING WITH A HIGH PERFORMANCE NETWORK INTERFACE**

(75) Inventors: **Shimon Muller**, Sunnyvale, CA (US); **Denton E. Gentry, Jr.**, Fremont, CA (US)

(73) Assignee: **Sun Microsystems, Inc.**, Santa Clara, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/260,324**

(22) Filed: **Mar. 1, 1999**

(51) **Int. Cl.**[7] **H04J 1/16**

(52) **U.S. Cl.** **370/230**; 370/235; 709/225; 709/228

(58) **Field of Search** 370/230, 231, 370/235, 392, 389, 225, 226, 241, 401, 428, 427, 473, 474, 394, 252, 466, 409; 709/225, 226, 235, 241, 228

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,414,704	A		5/1995	Spinney 370/60
5,583,940	A		12/1996	Vidrascu et al. 380/49
5,684,954	A		11/1997	Kaiserswerth et al. ... 395/200.2
5,748,905	A		5/1998	Hauser et al. 395/200.79
5,758,089	A		5/1998	Gentry et al. 395/200.64
5,778,180	A		7/1998	Gentry et al. 395/200.42
5,778,414	A		7/1998	Winter et al. 711/5
5,787,255	A		7/1998	Parlan et al. 395/200.63
5,793,954	A		8/1998	Baker et al. 395/200.8
5,870,394	A		2/1999	Oprea 370/392
5,920,705	A	*	7/1999	Lyon et al. 370/409
6,157,955	A	*	12/2000	Narad et al. 709/228

FOREIGN PATENT DOCUMENTS

EP	0 447 725	9/1991	..... G06F/15/16
EP	0 573 739	12/1993	..... H04L/12/56
EP	0 853 411	7/1998	..... H04L/29/06
EP	0 865 180	9/1998	..... H04L/12/56
WO	WO 95/14269	5/1995	..... G06F/7/08
WO	WO 97/28505	8/1997	..... G06F/13/14
WO	WO 99/00737	1/1999	..... G06F/13/00
WO	WO99/00945	1/1999	..... H04L/12/46
WO	WO99/00948	1/1999	..... H04L/12/56
WO	WO 99/00949	1/1999	..... H04L/12/56

OTHER PUBLICATIONS

Toong Shoon Chan, et al., "Parallel Architecture Support for High-Speed Protocol Processing," Feb. 1, 1997, *Microprocessors And Microsystems*, vol. 20, No. 6, pp. 325–339.

(List continued on next page.)

Primary Examiner—Wellington Chin
Assistant Examiner—William Schultz
(74) *Attorney, Agent, or Firm*—Park, Vaughan & Fleming LLP

(57) **ABSTRACT**

A system and method are provided for identifying related packets in a communication flow for the purpose of collectively processing them through a protocol stack comprising one or more protocols under which the packets were transmitted. A packet received at a network interface is parsed to retrieve information from one or more protocol headers. A flow key is generated to identify a communication flow that includes the packet, and is stored in a database of flow keys. When the packet is placed in a queue to be transferred to a host computer, the flow key and/or its flow number (e.g., its index into the database) is stored in a separate queue. Near to the time at which the packet is transferred to the host computer, a dynamic packet batching module searches for a packet that is related to the packet being transferred (i.e., is in the same flow) but which will be transferred later in time. If a related packet is located, the host computer is alerted and, as a result, delays processing the transferred packet until the related packet is also received. By collectively processing the related packets, processor time is more efficiently utilized.

27 Claims, 49 Drawing Sheets